

Open Source im Startup

Balázs Bárány

Linuxwochen Wien 2017

4. 5. 2017

Inhalt

Vorstellung: SCO2T

Architektur

Architektur

Datenbanken

Open-Source-Komponenten

Erfahrungen

Datenbanken

Pentaho

Traccar

Zusammenfassung

Über mich

Selbständiger Data Scientist – <https://datascientist.at>

Über mich

Selbständiger Data Scientist – <https://datascientist.at>

SCO2T – Roller-Sharing in Wien – <https://sco2t.com>



Vorstellung: SCO2T

Architektur

Architektur

Datenbanken

Open-Source-Komponenten

Erfahrungen

Datenbanken

Pentaho

Traccar

Zusammenfassung

1. Free-Floating Roller-Sharing in Wien

- ▶ Im Frühjahr 2015 gestartet
- ▶ Aktuell ca. 60 Fahrzeuge aktiv, ca. ein Drittel 125ccm
- ▶ Geschäftsgebiet: innerhalb des Gürtels bis zur Donau
- ▶ Reservierung und Miete über Webanwendung/App

1. Free-Floating Roller-Sharing in Wien

- ▶ Im Frühjahr 2015 gestartet
- ▶ Aktuell ca. 60 Fahrzeuge aktiv, ca. ein Drittel 125ccm
- ▶ Geschäftsgebiet: innerhalb des Gürtels bis zur Donau
- ▶ Reservierung und Miete über Webanwendung/App
- ▶ Privat finanzierte, unabhängige GmbH.
- ▶ 3 Gründer: Business, Marketing + Technik, IT/Analytik



Scooter

- ▶ 50-ccm- und 125-ccm-Modelle von Sym (Taiwan)
 - ▶ Elektro-Scooter?
- ▶ GPS-Box mit Anbindung an Bordelektronik
- ▶ Ferngesteuerte Aktivierung
- ▶ Schlüssel im Behälter unter Sitz
- ▶ 2 Helme







Vorstellung: SCO2T

Architektur

Architektur

Datenbanken

Open-Source-Komponenten

Erfahrungen

Datenbanken

Pentaho

Traccar

Zusammenfassung

Herausforderungen im Startup

- ▶ Fehlende Manpower
- ▶ Fehlende finanzielle Ressourcen
- ▶ Alles muß gleichzeitig geschehen
- ▶ Wenig Erfahrung

Herausforderungen im Startup

- ▶ Fehlende Manpower
- ▶ Fehlende finanzielle Ressourcen
- ▶ Alles muß gleichzeitig geschehen
- ▶ Wenig Erfahrung
- ▶ Lösungsmöglichkeiten
 - ▶ Auf Bewährtes zurückgreifen
 - ▶ Open Source, wenn möglich
 - ▶ (Fremdleistung einkaufen)

Komponenten

- ▶ Webanwendung (PHP)
- ▶ Backend für Betrieb und Analytik (Pentaho Business Analytics Platform)
- ▶ MySQL und PostgreSQL
- ▶ Traccar

Komponenten

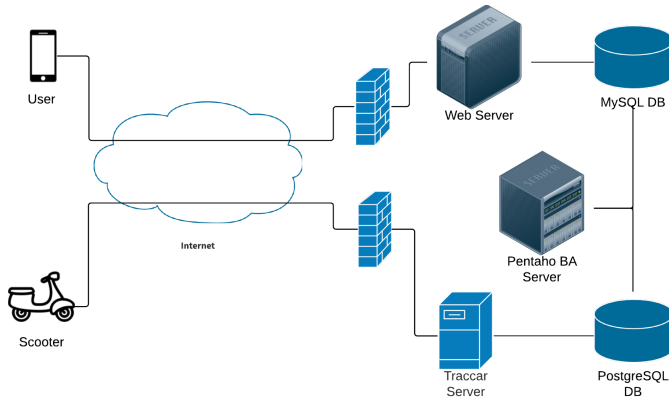
- ▶ Webanwendung (PHP)
- ▶ Backend für Betrieb und Analytik (Pentaho Business Analytics Platform)
- ▶ MySQL und PostgreSQL
- ▶ Traccar

- ▶ OwnCloud, Tiki CMS, OpenVPN, Git, ...

Komponenten

- ▶ Webanwendung (PHP)
- ▶ Backend für Betrieb und Analytik (Pentaho Business Analytics Platform)
- ▶ MySQL und PostgreSQL
- ▶ Traccar
- ▶ OwnCloud, Tiki CMS, OpenVPN, Git, ...
- ▶ Hosting in Wien bei österreichischer Firma

Architekturdiagramm



Details: Datenbanken

- ▶ MySQL
 - ▶ Web-App
- ▶ PostgreSQL
 - ▶ Backend für Traccar
 - ▶ Geoinformation
 - ▶ Analytik & Data Warehouse
- ▶ Verbindung über mysql_fdw
- ▶ PostgREST

PostGIS

- ▶ GIS-Erweiterung für PostgreSQL
 - ▶ Verarbeitung von Punkten, Linien und Flächen
 - ▶ Umrechnung zwischen Darstellungsformen und Projektionen
 - ▶ Berechnungen: Länge, Distanz, Fläche, ...
 - ▶ Verknüpfung, Aggregation geografischer Informationen mit SQL

PostGIS

- ▶ GIS-Erweiterung für PostgreSQL
 - ▶ Verarbeitung von Punkten, Linien und Flächen
 - ▶ Umrechnung zwischen Darstellungsformen und Projektionen
 - ▶ Berechnungen: Länge, Distanz, Fläche, ...
 - ▶ Verknüpfung, Aggregation geografischer Informationen mit SQL
- ▶ OpenStreetMap-Daten
- ▶ Open Government Data

Beispiele: PostGIS

OpenStreetMap importieren

```
osm2pgsql --host "$HOST" -U "$PGUSER" --database "$PGDB" \  
  --prefix "$PREFIX" \  
  --create --hstore --number-processes 2 \  
  --style import-osm.style \  
  --verbose --slim \  
  /tmp/osm.pbf
```

Beispiele: PostGIS

Länge der Fahrt aus Positionen

```
WITH points AS (  
    SELECT p.id, p.position  
    FROM positions p  
    WHERE ...  
)  
SELECT ST_Length(ST_MakeLine(position ORDER BY id)) AS ridelength  
FROM points;
```

Beispiele: PostGIS, Open Data

Ladezonen aus Open Data Wien

```
CREATE TABLE staging.wien_ladezonen_ogd (  
    FID text primary key,  
    OBJECTID integer not null unique,  
    ...  
);  
  
COPY staging.wien_ladezonen_ogd FROM PROGRAM 'curl  
"http://data.wien.gv.at/daten/..."'  
WITH csv DELIMITER ',' HEADER;
```

MySQL_FDW

- ▶ Foreign Data Wrapper für MySQL
 - ▶ Transparente Nutzung von MySQL-Tabellen
 - ▶ Einschränkungen: Datentypen, Syntax, Funktionalität
 - ▶ Echtzeit-Datensynchronisation mit Triggern
 - ▶ Komplexe Abfragen (Moderne SQL-Features)

Beispiel: MySQL_FDW

Anlegen der Foreign-Tabelle

```
CREATE SCHEMA prodsrv;
```

```
CREATE FOREIGN TABLE prodsrv.data_sync (  
    vehicle_id integer NOT NULL,  
    location character varying(250) NOT NULL,  
    ignition boolean  
)  
SERVER prodserver  
OPTIONS (dbname 'sco2t', table_name 'data_sync');
```

PostgREST

- ▶ Exportiert PostgreSQL-Schema als Web-API
 - ▶ JSON- oder CSV-Output
 - ▶ Lesen und schreiben, Funktionsaufrufe
 - ▶ PostgreSQL-Rollen und -Zugriffsrechte

PostgREST

- ▶ Exportiert PostgreSQL-Schema als Web-API
 - ▶ JSON- oder CSV-Output
 - ▶ Lesen und schreiben, Funktionsaufrufe
 - ▶ PostgreSQL-Rollen und -Zugriffsrechte
- ▶ Parkplatz-Warnung-API
 - ▶ View auf PostGIS-Abfrage
- ▶ Ankündigungen
 - ▶ View auf klassische Tabelle

PostgREST

- ▶ Exportiert PostgreSQL-Schema als Web-API
 - ▶ JSON- oder CSV-Output
 - ▶ Lesen und schreiben, Funktionsaufrufe
 - ▶ PostgreSQL-Rollen und -Zugriffsrechte
- ▶ Parkplatz-Warnung-API
 - ▶ View auf PostGIS-Abfrage
- ▶ Ankündigungen
 - ▶ View auf klassische Tabelle
- ▶ Eingebunden über Apache mod_proxy

Pentaho Business Analytics

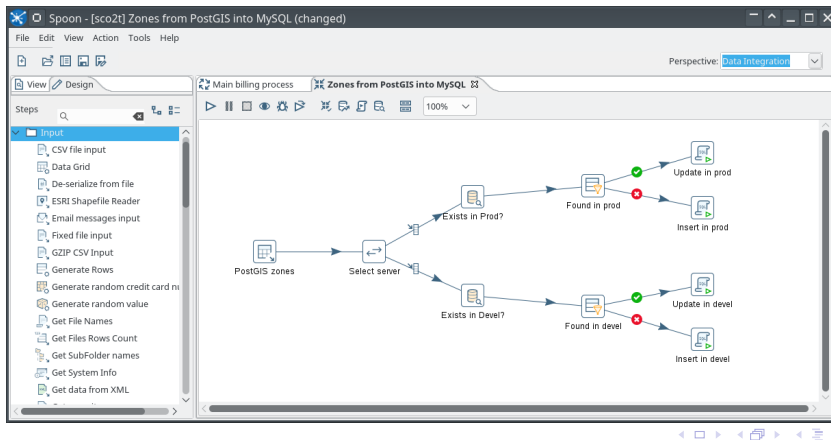
- ▶ Open-Source-Plattform für Analytik
 - ▶ Datenintegration
 - ▶ Reporting
 - ▶ Webserver für Dashboards und Online-Reports

Pentaho Business Analytics

- ▶ Open-Source-Plattform für Analytik
 - ▶ Datenintegration
 - ▶ Reporting
 - ▶ Webserver für Dashboards und Online-Reports
- ▶ Anwendungen:
 - ▶ Dashboards für Analyse und Betrieb
 - ▶ Datenaustausch und Automatisierung
 - ▶ Berichte und Benachrichtigungen
 - ▶ Rechnungslegung

Open-Source-Komponenten

Beispiel: Pentaho Data Integration



Traccar

- ▶ Open-Source-Plattform fürs Tracking von GPS-Geräten
 - ▶ Web-Karte, Benutzerverwaltung etc.
 - ▶ Unterstützt viele Protokolle
 - ▶ Schwerpunkt: Fahrzeuge

Analytik

- ▶ Deskriptive Analyse
 - ▶ SQL
 - ▶ Pentaho-Dashboards

Analytik

- ▶ Deskriptive Analyse
 - ▶ SQL
 - ▶ Pentaho-Dashboards
- ▶ Predictive Analytics
 - ▶ RapidMiner Studio & RapidMiner Server
 - ▶ R mit RStudio, in PostgreSQL mit PL/R

Vorstellung: SCO2T

Architektur

Architektur

Datenbanken

Open-Source-Komponenten

Erfahrungen

Datenbanken

Pentaho

Traccar

Zusammenfassung

Datenbanken

- ▶ MySQL
 - ▶ Tabellentypen
 - ▶ Sperren
 - ▶ Rechtevergabe
 - ▶ Spaltentypen: Zeit, Text, ...
 - ▶ Funktionalität

Datenbanken

- ▶ MySQL
 - ▶ Tabellentypen
 - ▶ Sperren
 - ▶ Rechtevergabe
 - ▶ Spaltentypen: Zeit, Text, ...
 - ▶ Funktionalität
- ▶ PostgreSQL
 - ▶ Skalierbar und robust
 - ▶ PostGIS: essenziell
 - ▶ mysql_fdw: gut, Einschränkungen von MySQL
 - ▶ PostgREST: gute Idee, gut ausgeführt

SQL vs. NoSQL

- ▶ Grundproblem: Ökosystem
 - ▶ SQL ist Standard – alles darauf angepaßt
 - ▶ Bei NoSQL keine fertigen Lösungen
 - ▶ Tools, Reporting, Backup, Frontends, ...

SQL vs. NoSQL

- ▶ Grundproblem: Ökosystem
 - ▶ SQL ist Standard – alles darauf angepaßt
 - ▶ Bei NoSQL keine fertigen Lösungen
 - ▶ Tools, Reporting, Backup, Frontends, ...
- ▶ Performance
 - ▶ Standardfall in SQL-Datenbanken bestens optimiert
 - ▶ Verteilte Lösungen skalieren mit NoSQL-Datenbanken
 - ▶ Overhead für Ein-Server-Betrieb
 - ▶ Datenmenge muß erst da sein

SQL vs. NoSQL

- ▶ Grundproblem: Ökosystem
 - ▶ SQL ist Standard – alles darauf angepaßt
 - ▶ Bei NoSQL keine fertigen Lösungen
 - ▶ Tools, Reporting, Backup, Frontends, ...
- ▶ Performance
 - ▶ Standardfall in SQL-Datenbanken bestens optimiert
 - ▶ Verteilte Lösungen skalieren mit NoSQL-Datenbanken
 - ▶ Overhead für Ein-Server-Betrieb
 - ▶ Datenmenge muß erst da sein
- ▶ Funktionalität
 - ▶ SQL enorm mächtig (CTEs, Window Functions)
 - ▶ NoSQL: keine Standard-Abfragesprache
 - ▶ Komplexe Abfragen nur mit Programmierung

Pentaho Data Integration

- ▶ Grafische Erstellung komplexer Datenverarbeitungsprozesse
- ▶ Vielfältige Input- und Output-Komponenten
- ▶ Integration mit Reporting, Web, Pentaho Business Analytics
- ▶ Zuverlässig und performant
 - ▶ Java: Start, Speicherverbrauch
- ▶ Ausführung am Server aus Cron

Pentaho Business Analytics Server

- ▶ Komplexe Java-Serveranwendung
- ▶ Performance: behäbig, speicherfressend
- ▶ Kleine Bugs/misfeatures

Pentaho Business Analytics Server

- ▶ Komplexe Java-Serveranwendung
- ▶ Performance: behäbig, speicherfressend
- ▶ Kleine Bugs/misfeatures
- ▶ Dashboards sehr flexibel
- ▶ Dashboard-Erstellung: halb grafisch, HTML und JavaScript
- ▶ Alternative zu Cron

Traccar

- ▶ Robust und zuverlässig
- ▶ Datenbankschema-Änderung zwischen Versionen

Vorstellung: SCO2T

Architektur

Architektur

Datenbanken

Open-Source-Komponenten

Erfahrungen

Datenbanken

Pentaho

Traccar

Zusammenfassung

Zusammenfassung

- ▶ Startup-IT mit geringen Kosten
 - ▶ Kaum Kompromisse

Zusammenfassung

- ▶ Startup-IT mit geringen Kosten
 - ▶ Kaum Kompromisse
- ▶ Entwicklung mit wenig Zeit- und externem Entwickler-Einsatz

Zusammenfassung

- ▶ Startup-IT mit geringen Kosten
 - ▶ Kaum Kompromisse
- ▶ Entwicklung mit wenig Zeit- und externem Entwickler-Einsatz
- ▶ Komponenten und Architektur in der Praxis bewährt

Fragen?

- ▶ Balázs Bárány, <balazs@tud.at>
- ▶ <https://datascientist.at/>
- ▶ <https://sco2t.com/>